Cells Segmentation Using the Hybrid of Image Morphology and Edge Detector Algorithm and Cell Counting

Feng-Ming Li

Department of Electrical Engineering and Computer Science

Case Western Reserved University, Cleveland, OH, Email: Fengming.li@case.edu

Abstract--In recent decades, demand for applying image-processing techniques of improving biomedical diagnosis has been raising rapidly because traditional system process does not only cost too much time and human resources but also causes a lot of errors and unexpected results. In addition, traditional method requires skilled experts to work on it so that only minority can be qualified. As an objective and programmed method is required to eliminate those shortcomings discussed above. In this paper, I propose the hybrid of Image Morphology and Canny Edge detector Algorithm (HIMCA) to segment, count, and locate sensors in retina automatically. In addition, I also offer a feedback control scheme to correct the errors HIMCA makes. HIMCA can improve the performance of cell segmentation a lot although there are still some cells hard to tell. However, it has a good chance to solve this problem in the near future.

Key words--Cell counting, image morphology, morphology, edge detector, segmentation.

Ⅰ Introduction

Cell detection is a critical issue for clinicians. Since the early 1790s computer-assisted image analysis started to contribute significantly toward the solution of these problems, particulr--ly in the field of cytology automation. The research in automating the analysis of cytological specimens was the main drive for exploring this area, and cell segmentation and cell counting is a very common task. However, cell counting, in histological preparations containing histological noise, such as debris and synthetic materials, is arduous to automate using standard image analysis tools independently. In addition, manual cell counting is a laborious task suffering from subjectivity and an inability to derive more complex information such as stain distributions, geometric relationships to other important structures such as vascular networks. Moreover, the biomedical images captured by the medical cameras are usually vague, and it wastes too much time and money to retake new images. Therefore appropriate image processing skills are required. In this paper, our first goal is to calculate the number of cells. But before the imager is ready, there are still many delicate processing required, which will be discussed in the following sections.

Ⅱ Techniques

In this section, I will describe techniques I used in this paper.

A. Second-Order Edge Detection

Many algorithms have been developed to

segment images using edge detection [1]. One of the approaches presented in this paper utilizes second-derivative edge detection based on identifying the generated zero crossings. In this case, I tried to use two edge detectors, which are described in the following:

(1) Laplacian of Gaussian (LoG)

The LoG edge detector [2] has the advantage of smoothing prior to edge detection (crucial in noisy histological material), and its zero-crossing response when an edge is encountered enhances its effectiveness. The concept behind LoG is based on convolving an image with the Laplacian of a 2D Gaussian function, that is, a rotationally symmetric convolution filter:

$$LoG(x,y) = \nabla^2 G(x,y) = A(2 - \frac{x^2+y^2}{\sigma^2})e^{(x^2+y^2)/2\sigma^2} \tag{1}$$

where A is simply a constant used to regulate the magnitude of the Gaussian, is the standard deviation determining its spatial scale, and x, y are the 2D spatial coordinates of the image.

(2) Canny

In this paper, I also utilize canny edge detector. Canny edge detector [3] uses a filter based on the first derivative of a Gaussian, and it is sensitive to noise present on raw unprocessed image data. Therefore, to begin with the raw image is convolved with a Gaussian filter. The result is as a blurred version of the original which is not affected by noisy pixels to any significant degree. The Canny operator was considered to be an optimal edge detector. It takes as input a gray scale image, and produces as output an image showing the positions of tracked intensity discontinuities.

B. Histogram equalization

Histogram equalization is a method in image processing of contrast adjustment using the image's histogram. Consider a discrete grayscale image, and n_i is the number of occurrences of gray level i. The probability of an occurrence of a pixel of level i in the image is

$$P(x_i) = \frac{n_i}{n}, i \in 0,\ldots,L-1 \tag{2}$$

L is the total number of gray levels, n is the total number of pixels, and p is the image's histogram, normalized to [0,1]. C is the cumulative distribution function defined by:

$$c(i) = \sum_{j=0}^{i} p(x_i) \tag{3}$$

A transformation of the form y = T(x) that will produce a level y for each level x in the original image, such that the cumulative probability function of y will be linearized across the value range. In order to map the values back into their original domain, equation (4) needs to be applied on the result:

$$y_i' = y_i \bullet (\max - \min) + \min \tag{4}$$

C. Erosion

Erosion is one of the algorithms to thin or shrink objects. The mathematical definition of erosion of A by B is the set of all structuring element of origin locations where the translated B has no overlap with the background of A.

$$A \ominus B = \{z|(B) \cap Ac \neq 0\} \tag{5}$$

D. Conditional dilation

The condition dilation operator [4] involves two images, the input image and the conditional image, and a structuring element. The resultant image is the intersection between the input image dilated with the structuring element and the conditional image. In this way, the condition image acts like a mask on the dilation. The mathematical formulation can be written as

$$A \oplus |IB=(A \oplus B)\cap I \qquad (6)$$

E. Otsu's method

Otsu's algorithm [5] is used to perform thresholding, or, the reduction of a gray level image to a binary image. The algorithm assumes the image histogram is bimodal and finds the optimum threshold separating the two classes. It is capable of separating a foreground from a background. Otsu's method is defined by the following equation:

$$\sigma(t) = q_1(t)\sigma_1(t) + q_2(t)\sigma_2(t) \qquad (7)$$

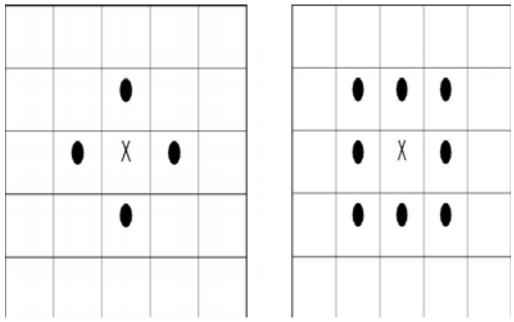
σ(t) is the sum of weighted variances of the two modes of the histogram as a function of the threshold t. σ(t) and q(t) are the respective variance and probability of one of the two modes of the histogram separated by a threshold t. The threshold t that results in the minimization of σ(t) produces the desired, "foreground and background," bi-level image.

E. Watershed Segmentation

Watershed segmentation [6] finds pixels at the midpoints between features by treating the pixel values as topographic elevations. The areas between features become catchment basins which depict the influence zones in the image.

F. Objects Counting Method

Once a gray level image has been processed to remove noise and thresholded to produce a binary image, a connected components labeling operator can be employed to group the binary-1 pixels into maximal connected region. These regions are called the connected components of the binary image, and the associated operator is called the connected components operator. Figure 1 illustrates the connected components operator as applied to the 1-pixels of a binary image.



(a) (b)

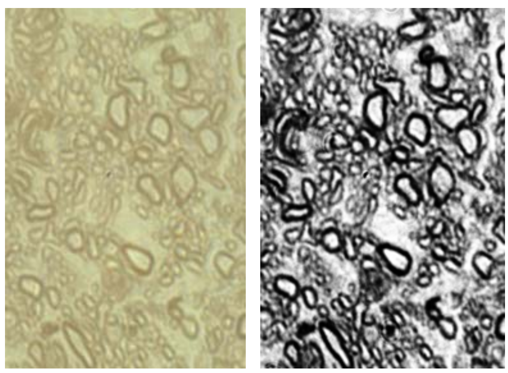
Fig1 (a) Function of the connected components operator on a binary image; (b) Symbolic image produced from (a) by the connected components operator.

Ⅲ Algorithm and Results

In this section, I will describe the basic operation of this algorithm.

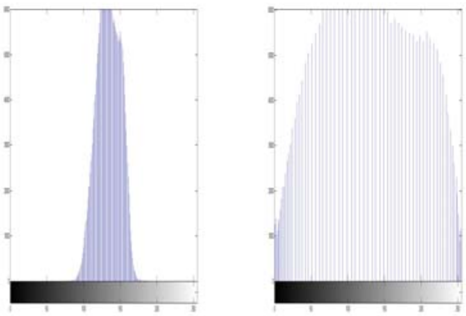
A. Image Pre-Processing

First, we have to notice the characteristics of the image which is going to be processed. In this case, we have to count the number of cells of the image, which is shown in Figure 2. In Figure 3,



(a) (b)

Fig 2 (a) The original image of the retina of mouse. (b) The image after histogram equalization.



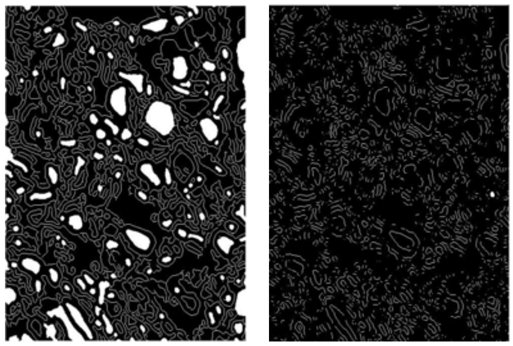
(a) (b)

Fig3 (a) Histogram of the original image (Red) (b) The histogram after being processed by histogram equalization

the image in figure 3(a) illustrates the contrast of original image is not high. Therefore, we have to utilize some techniques, such as power series and histogram equalization, to adjust the intensity values. In this paper, I used both histogram equalization and power series method in different situation. The right-hand-side images of Figure 2 and Figure3 illustrate the result used by histogram equalization.

B. Isolating cells out and Counting

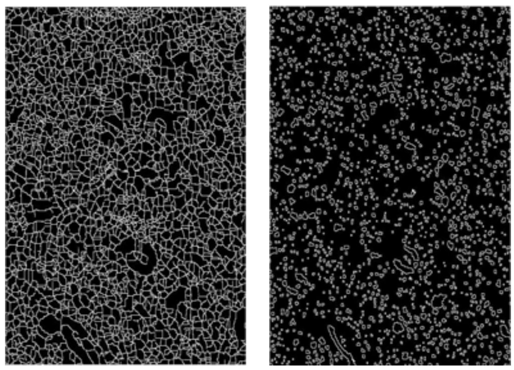
Since we have to count number of cells, we have to isolate cells out of background. There are many methods, such as segmentation, designing Frequency filter, morphological processing, watershed segmentation algorithm and so on. In this paper, I compared different methods and develop a new algorithm to solve this problem. First, Laplacian of Gaussian Method and Canny edge detector are utilized to detect the edge. Then the objects which have continuing and surrounding perimeter are filled. Figure 4 shows that Canny edge detector has a better result. However, it is not good enough because there are still many cells which are not counted in yet. Obviously, both of these methods cannot meet our requirement. Figure 5 illustrates



(a) (b)

Fig4 (a) Cells isolated out by Canny algorithm. (b) Cells isolated out by LoG.

the result calculated by Watershed segmentation. Again, the performance is not good. In Figure 5(a), over segmentation is acute, and the image

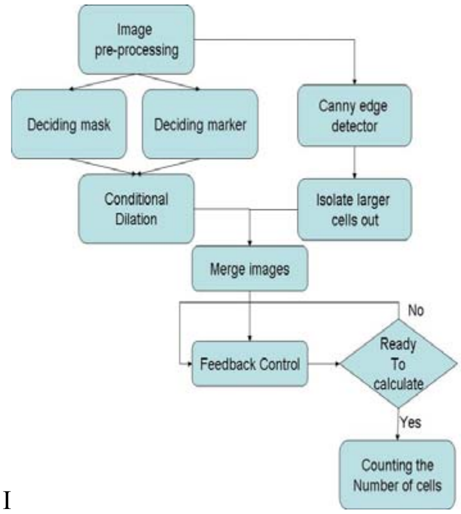


(a) (b)

Fig 5 (a) The image processed by watershed segmentation directly (b) The image processed by watershed segmentation after blurring image.

in Figure 5(b), which is blurred by image processing skill to reduce the effect of over segmentation, is still awful.

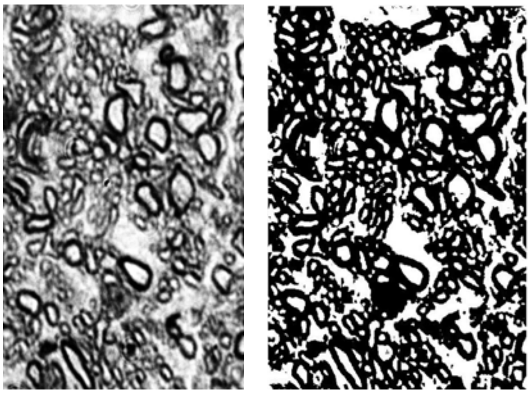
According to the discussion above, we need to offer another algorithm to solve this problem. Here, I provide a better algorithm to segregate cells. My strategy is shown in Fig6. First, I transformed the original RGB image to gray level image. Secondly, because calculating only threshold value of the image will result in losing some vague small cells. I cut the image into 3x5 pieces, and then I used Otsu's method to calculate every threshold value of every piece.



I

Fig 6. Flow chart of HIMCA

Thereafter, the binary image of each piece will be produced by thresholding it with its specific threshold values. Figure 7 illustrates the results.



(a) (b)

Fig 7 (a) The image before thresholding by Otsu's Algorithm. (b) The image after process by thresholding every piece (3x5) of the image using Otsu's Algorithm

However it is tricky to decide how many pieces you want to segment. If the number of pieces you choose is too large, a lot of cells will be cut open, and a lot of cells will be merged if the number of pieces you choose is too small instead. Figure8 illustrates this problem.

Thirdly, the cells should be secluded from the boundary. In this case, I adopted the following steps: (1) I used the binary image in Figure 7(b) to be the mask. (2) Since the cells have to be isolated, and most of them are smaller and have more regular shape than the background, eroding

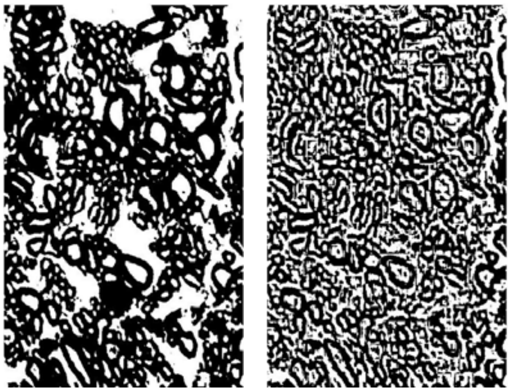
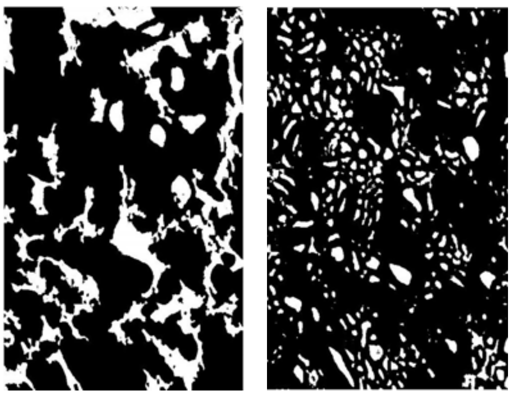


Fig 8 (a) Cutting the image into 3x3 pieces (b) Cutting the image into 50x50 pieces

and recognition image processing can be used to identify the cells and separate them from background. Here I eroded the smaller objects, and reserved the bigger blocks, then the processed image can be built for the marker. Thereafter, I used conditional dilation to reconstruct the binary image. The image in Figure 9(a) illustrates the objects which are considered as background, and the image in



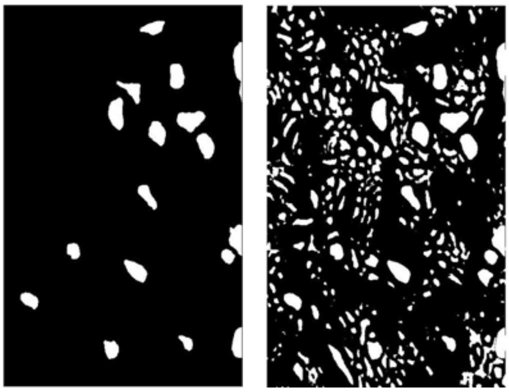
(a) (b)

Fig 9 (a) Objects considered as background. (b) Objects considered as cells.

Figure 9(b) shows the objects considered as cells instead. As you can see, the background objects are totally separated, but some bigger cells are picked out, too. Clearly, we have to give some feedback for this image to pick these cells out.

Recalling that the cells we secluded by using Canny Algorithm, it can be noticed that most of them are bigger cells with clear edge which fortuitously are those larger cells missing in Figure 9(b). Here, we can erode those thin lines in Figure 4(a) by using conditional dilation. Then we can get the useful image as Figure 10(a) shows. Therefore, it is reasonable to merge these two images to be prepared for cell counting, which is shown in Figure 10(b).

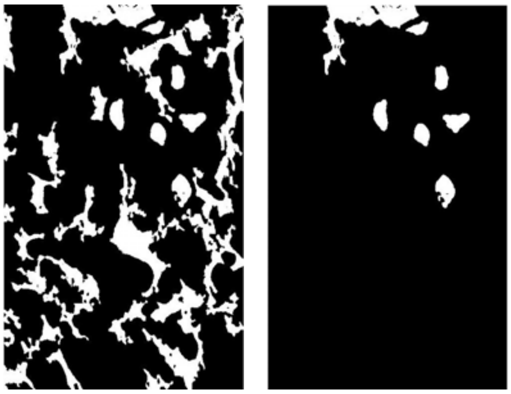
Since sometimes there are some objects which are difficult to be identified as background or cells, it always happen to distribute them into wrong group. Therefore, I offered a feedback control to eliminate those objects which you considered them as being in the wrong group. The principle is simple, and the only thing you need to do is to point the objects which you thi-



(a) (b)

Fig 10 (a) Cells isolated by Canny Algorithm. (b) The image merged Fig 10(a) with Fig 9(b)

-nk they are not belong to this group. Then use the conditional dilation to seclude them so that you can get the binary image with these specific cells. Figure 11 illustrates the example of this method. As you can see, you can choose any objects that you want to rule them out no matter what characteristics they have. In other words,



(a) (b)

Fig 11 (a) Example of image which is prepared for secluding objects out manually. (b) Image of objects which are selected to be isolated.

we can identify these objects if they belong to the right group after they are being distributed by the algorithm manually so that the performance can be improved. Now, the image should be prepared to be calculated the number of cells. In this paper, every eight-connected object is considered as one independent one, and according to the counter, the number is 406. Figure 12 illustrates the output image and every object which is considered as cells by the algorithm is labeled. As you can see, the algorithm does a good job. However, there are still some cells, which have vague edge, missed, and some cells with both clear edge and vague edge are over counted. These problems should be solved in the near future, and it is not included in this paper.

Ⅳ Conclusion and Future Work

From the discussion above, it can be noticed that it is arduous to use any standard image analysis

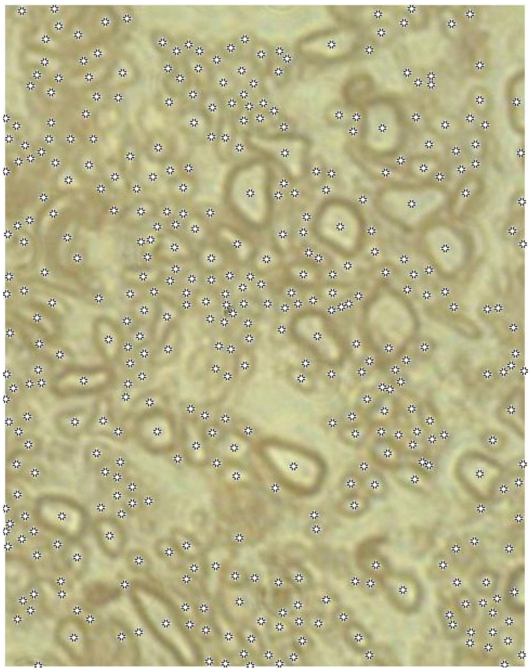


Fig 12 Output image with cells labeled.

tools independently to analyze this problem. However, HIMCA offered in this paper can improve the performance sharply. Although there are still some problems left, such as over counting and cells missed, these problems have a good chance to be solved by two methods. One of them is the feedback control scheme provided in this paper, and another one is to deal with the areas which have vague edge or half-vague and half-clear edge locally. The future work will be developing the algorithm to detect the law contrast area, and to strengthen the contrast automatically so that automatic cell counting will be more dependable.

References

[1] Pal NR, Pal SK. A review on image segmentation techniques. Pattern Recognition 1993;26:1277–1294.

[2] Marr D, Hildreth EC. Theory of edge detection. Proc R Soc London 1980;B207:187–217.

[3] Canny, J., A Computational Approach To Edge Detection, IEEE Trans. Pattern Analysis and Machine Intelligence, 8:679-714, 1986

[4] Vincent, L., "Morphological Grayscale Reconstructi--on in Image Analysis: Applications and Efficient Algorithms," IEEE Transactions on Image Processing, Vol. 2, No. 2, April, 1993, pp. 176-201.

[5] Otsu, N., "A Threshold Selection Method from Gray-Level Histograms," IEEE Transactions on Systems, Man, and Cybernetics, Vol. 9, No. 1, 1979, pp. 62-66.

[6] Vincent, Luc, and Pierre Soille, "Watersheds in Digital Spaces: An Efficient Algorithm Based on Immersion Simulations," IEEE Transactions of Pattern Analysis and Machine Intelligence, Vol. 13, No. 6, June 1991, pp. 583-598.